

(19)



JAPANESE PATENT OFFICE

PATENT ABSTRACTS OF JAPAN

JPA07-114638

(11) Publication number: **07114638 A**

(43) Date of publication of application: **02.05.95**

(51) Int. Cl

**G06T 5/20**

(21) Application number: **05281827**

(71) Applicant: **RICOH CO LTD**

(22) Date of filing: **15.10.93**

(72) Inventor: **TACHIKAWA MICHİYOSHI**

(54) **IMAGE PROCESSOR**

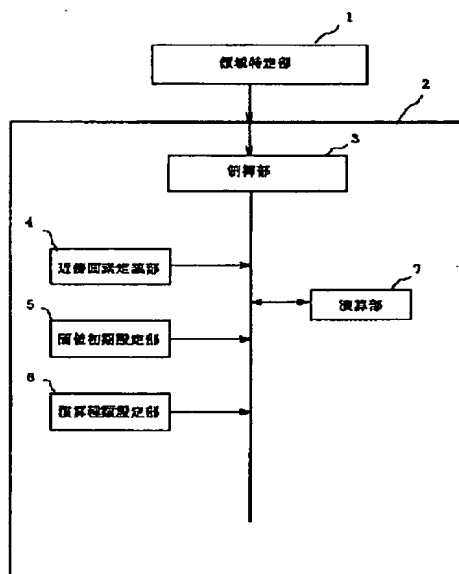
(57) Abstract:

PURPOSE: To repair and estimate an image at a high speed with high quality by processing only pixels in a specific area in an image.

CONSTITUTION: An image process part 2 while determining the value of a specific unknown pixel in an area to be processed by using the values of known pixels when the number of the known pixels having the known values among pixels nearby the specific unknown pixel in the area to be processed is larger than a specific area leave the unknown pixel as it is without determining its value when the number of the known pixels is smaller than the specific threshold value, regards the unknown pixel whose value is determined as a known pixel after processing the respective pixels in the area to be processed with the specific threshold value, and repeats similar processes for the remaining unknown pixels. When those processes are performed, a threshold value initial setting part 5 sets the initial value of the threshold value to an

optimum value according to the size of the area to be processed and/or the shape of the area to be processed.

COPYRIGHT: (C)1995,JPO





JPA 7-114638

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平7-114638

(43) 公開日 平成7年(1995)5月2日

(51) Int. Cl. <sup>6</sup>

識別記号

庁内整理番号

F I

技術表示箇所

G06T 5/20

9191-5L

G06F 15/68

410

審査請求 未請求 請求項の数7 F D (全18頁)

(21) 出願番号 特願平5-281827

(22) 出願日 平成5年(1993)10月15日

(71) 出願人 000006747

株式会社リコー

東京都大田区中馬込1丁目3番6号

(72) 発明者 立川 道義

東京都大田区中馬込1丁目3番6号 株式  
会社リコー内

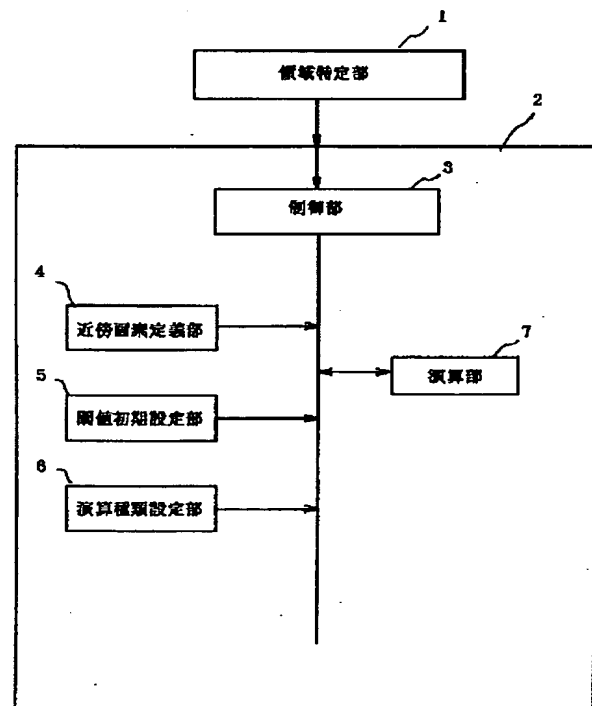
(74) 代理人 弁理士 植本 雅治

(54) 【発明の名称】 画像処理装置

(57) 【要約】

【目的】 画像中の所定領域内の画素についてのみ処理を行い、画像を高速にかつ高品質に修復、推定できる。

【構成】 画像処理部2は、処理対象領域内の特定の未知画素の近傍画素のうちで既知の値をもつ既知画素の個数が所定閾値よりも大きいときに既知画素の値を用いて処理対象領域内の特定の未知画素の値を決定する一方、既知画素の個数が所定閾値よりも小さいときには、未知画素の値を決定せず未知画素のまま残し、処理対象領域内の各画素について所定閾値で上記処理を行った後、値の決定された未知画素を既知画素とし、また、閾値を大きい値から小さい値に段階的に変更し、残されている未知画素に対して同様の処理を繰返し行う。このような処理を行うに際し、閾値初期設定部5は、処理対象領域の大きさ、および/または、処理対象領域の形状に応じ、上記閾値の初期値を最適なものに設定する。



## 【特許請求の範囲】

【請求項1】 画像中の処理対象領域に含まれる画素に対して所定の画像処理を施す画像処理装置であって、前記画像処理装置は、前記処理対象領域の大きさに関する情報、および／または、処理対象領域の形状に関する情報に基づいて、閾値の最適な初期値を決定する閾値初期決定手段と、前記処理対象領域内の各画素が当初、完全に未知の値あるいは不確かな値をもつ未知画素であるとし、処理対象領域内の特定の未知画素に着目するとき、該未知画素の近傍画素のうちで既知の値をもつ既知画素の個数が所定の閾値よりも大きい場合に限り、近傍画素のうちの既知画素の値を用いて未知画素の値を決定する一方、既知画素の個数が所定の閾値よりも小さいときには、未知画素の値を決定せず未知画素のまま残すように演算処理を行なう演算処理手段と、処理対象領域内の各画素について、前記閾値初期決定手段によって決定された閾値の最適な初期値のところから前記演算処理手段による演算処理を行なわせ、処理対象領域内の各画素について演算処理がなされた後、値の決定された未知画素を既知画素とし、また、閾値を変更し、残されている未知画素に対して前記演算処理を繰り返し行なう制御手段とを備えていることを特徴とする画像処理装置。

【請求項2】 請求項1記載の画像処理装置において、前記閾値初期決定手段は、前記処理対象領域の大きさに関する情報を、処理対象領域内の画素数として検出するようになっていることを特徴とする画像処理装置。

【請求項3】 請求項1記載の画像処理装置において、前記閾値初期決定手段は、前記処理対象領域の大きさに関する情報を、画像全体の画素数に対する処理対象領域内の画素数の割合として検出するようになっていることを特徴とする画像処理装置。

【請求項4】 請求項1記載の画像処理装置において、前記閾値初期決定手段は、前記処理対象領域の形状に関する情報を、処理対象領域を矩形状に切出し、切出した矩形の縦方向と横方向の長さの比に基づき、検出するようになっていることを特徴とする画像処理装置。

【請求項5】 請求項1記載の画像処理装置において、前記閾値初期決定手段は、前記処理対象領域の形状に関する情報を、処理対象領域を矩形状に切出し、切出した矩形内の画素数と処理対象領域内の画素数との比に基づき、検出するようになっていることを特徴とする画像処理装置。

【請求項6】 請求項1記載の画像処理装置において、前記閾値初期決定手段は、前記処理対象領域の形状に関する情報を、処理対象領域を矩形状に切出し、切出した矩形の縦方向と横方向の長さの比と、切出した矩形内の画素数と処理対象領域内の画素数との比とに基づき、検出するようになっていることを特徴とする画像処理装置。

【請求項7】 請求項1記載の画像処理装置において、

前記閾値初期決定手段は、前記処理対象領域の大きさが大きい程、閾値の初期値を大きく決定し、また、前記処理対象領域の形状が細長い程、閾値の初期値を大きく決定するようになっていることを特徴とする画像処理装置。

## 【発明の詳細な説明】

【0001】

【産業上の利用分野】 本発明は、画像の修復などに用いられる画像処理装置に関する。

【0002】

【従来の技術】 従来、パターン認識などの分野において、例えば原稿上のしみや汚れ、あるいは損傷の影響を除去するため、読み取った画像データに対して前処理が施される。この種の前処理技術の代表的なものとして、例えば著者「森 俊二」による文献「“画像認識の基礎〔I〕”、オーム社、PP24～35、1986年」に示されているような、k最近傍法や、k最近傍法を発展させた選択平均法、近傍加重平均法等がある。例えば、k最近傍法は、着目する画素の近傍に所定の大きさのマスク（例えば3×3の大きさのマスク）を設定し、このマスクに含まれる近傍点（3×3の大きさのマスクでは8個の近傍点）のうちから一定個数k（kは固定）の近傍点を選択し、k個の近傍点の平均をとり、その平均値によって、着目する画素、すなわち近傍の中心の値を置き換えるようになっており、これによって、画像上の孤立点（原稿上のしみなどによる影響）を除去することができる。

【0003】

【発明が解決しようとする課題】 しかしながら、従来では、画像の補正を行なうのに、例えば前述したマスクを画像の左上から右下に向けて順次に移動し、画像中の欠陥を生じていない画素についても、一率に平均化処理を行っていた。このため、相当の処理時間を要し、また、補正を行なう必要のない画素についてまで処理がなされてしまうので、全体の画像品質が低下してしまうなどの問題があった。

【0004】 さらに、従来では、画像中に生じた任意の形状、大きさの欠陥について、これを良好に修復することができないという欠点があった。

【0005】 本発明は、画像中の所定領域内の画素についてのみ修復や推定などの処理を行なうことができ、画像を高速にかつ高品質に修復、推定等することの可能な画像処理装置を提供することを目的としている。

【0006】

【課題を解決するための手段および作用】 上記目的を達成するために、請求項1乃至請求項7記載の発明は、処理対象領域内の各画素が当初、完全に未知の値あるいは不確かな値をもつ未知画素であるとし、処理対象領域内の特定の未知画素に着目するとき、該未知画素の近傍画素のうちで既知の値をもつ既知画素の個数が所定の閾値

よりも大きい場合に限り、近傍画素のうちの既知画素の値を用いて未知画素の値を決定する一方、既知画素の個数が所定の閾値よりも小さいときには、未知画素の値を決定せず未知画素のまま残すようになっており、処理対象領域内の各画素について所定の閾値での前記処理がなされた後、値の決定された未知画素を既知画素とし、また、閾値を変更し、残されている未知画素に対して処理を繰り返し行なう画像処理において、上記閾値の最適な閾値を、処理対象領域の大きさに関する情報、および／または、処理対象領域の形状に関する情報に基づいて決定するようになっている。これにより、処理対象領域の大きさ、形状に応じた最適な処理を行なうことができ、画像を高速にかつ高品質に修復、推定等することができ

【0007】

【実施例】以下、本発明の実施例を図面に基づいて説明する。図1は本発明に係る画像処理装置の構成例を示す図である。図1を参照すると、この画像処理装置は、図2に示すように、画像PC中において処理対象(例えば修復対象)となる領域PCUを特定する領域特定部1と、領域特定部1によって特定された処理対象領域PCUに含まれる画素に対して所定の画像処理を施す画像処理部2とを備えている。

【0008】ここで、画像処理部2は、領域特定部1によって特定された処理対象領域PCUに含まれる各画素が当初、完全に未知の値あるいは完全には未知でないが不確かな値をもつ画素(以後、これを未知画素と称す)であり、この処理対象領域PCU外の画像PCの領域PCKの各画素が全て既知の値をもつ既知画素であるとし、処理対象領域PCU内の未知画素の値を、既知領域PCKの既知画素の値を用いて既知の(確かな)ものに定めるようになっている。すなわち、画像処理部2は、画像PC中の全ての画素に対して処理を行なうのではなく、画像PC中の処理対象領域PCU内の画素のみに着目し、処理対象領域PCU内の画素のみにに対して処理を行なうようになっている。

【0009】また、画像処理部2は、処理対象領域PCU内の特定の未知画素に対する処理を行なうのに、該特定の未知画素の近傍の画素に着目し、近傍画素のうちで既知の値をもつ既知画素を用いて、該特定の未知画素の値を定める(既知のものとする)ようになっている。この際、画像処理部2は、処理対象領域PCU内の特定の未知画素の近傍画素のうちで既知の値をもつ画素の個数を計数し、既知の値をもつ画素の個数が所定の閾値よりも大きい場合に限り、近傍画素のうちの既知の値をもつ既知画素の値を用いて未知画素の値を決定する一方、既知の値をもつ画素の個数が所定の閾値よりも小さいときには、未知画素の値を決定せずに未知画素のまま残すようにしている。そして、処理対象領域PCU内の各画素について所定の閾値で上記処理がなされた後、値の決定さ

れた未知画素を既知画素とし、また、閾値を大きい値から小さい値に段階的に変更し、まだ値の定まっていない残されている未知画素に対して同様の処理を繰り返し行なうようになっている。

【0010】このような処理機能を実現するため、図1の構成例では、画像処理部2には、処理全体の制御を行なう制御部3と、近傍画素を定義するための近傍画素定義部4と、閾値の初期値を設定するための閾値初期設定部5と、演算の種類を設定するための演算種類設定部6と、演算種類設定部6に設定された種類の演算を行なう演算部7とが備わっている。

【0011】図3はこの画像処理部2の処理例を示すフローチャートである。なお、以下では、説明を簡単にするため、一例として、処理対象領域PCU内の各画素に対し、1画素単位に処理がなされ、また、演算部7では、処理対象領域PCU内の未知画素を完全に未知の値のものと仮定し(未知画素が不確かな値をもっているものを一切考慮せずに)、近傍画素のうちの既知画素の値だけに基づいて、未知画素の値を決定する処理がなされるものとする。

【0012】画像処理部2が実際の処理を開始する前提として、画像中において、未知画素からなる処理対象領域PCUと、処理対象領域PCU以外の既知画素からなる既知領域PCKとが、領域特定部1により特定されており、各画素の存在する位置(例えばメモリ内位置)がわかっている必要がある。なお、このとき、処理対象領域PCU内の各画素を特定するため、各画素に番号jが付されているとし、また、処理対象領域PCU内の画素の総数Nがわかっているとする。

【0013】また、近傍画素定義部4により、1つの未知画素に対する近傍画素が予め定義されている必要がある(例えば、近傍の形状、大きさなど)。また、演算種類設定部6により処理演算の種類(例えば近傍画素のうちの既知画素の値の平均値をとるなどの演算の種類)が設定されている必要がある。また、閾値初期設定部5により閾値kの初期値が設定されている必要がある。

【0014】すなわち、近傍画素の定義、演算の種類、閾値の初期値が、処理対象となる画像の種類(例えば、画像が2値画像であるか多値画像であるか、あるいは、画像が文書画像であるか、写真画像であるか、絵図画像であるかなど)に応じて、あるいは、処理対象領域の種類(処理対象領域の大きさ、形状など)に応じて、適切なものに設定されている必要がある。

【0015】このような前提が整った後(ステップS1)、jを“1”に初期設定する(ステップS2)。次いで、画像処理部2は、処理対象領域PCUからj番目の画素を取り込み(ステップS3)、この画素の近傍画素のうちで既知の値をもつ既知画素の個数を計数する(ステップS4)。そして、その計数値CNTを閾値kと比較し(ステップS5)、計数値CNTが閾値kと同じか閾値

kよりも大きいときには、演算部7は、j番目の画素の近傍画素のうちで既知の値をもつ既知画素に基づき、j番目の画素の値を定める(ステップS6)。演算の種類が例えば“平均値”であるときには、既知画素の値の平均値を求め、これをj番目の画素の値として決定する。これに対し、ステップS5において、計数値CNTが閾値kよりも小さいときには、演算を行わず、j番目の画素の値を未知のまま残す(ステップS7)。

【0016】j番目の画素についてステップS6またはステップS7の処理を行なった後、jを“1”だけ歩進して(ステップS8)、再びステップS3に戻り、処理対象領域PCU内の次の画素について同様の処理を繰り返し行なう。処理対象領域PCU内の各画素について順次にこのような処理を繰り返し行ない、処理対象領域PCU内の最後の画素についての処理を終了すると(ステップS9)、各画素についての処理結果によって、処理対象領域PCU内の未知画素の値を書き換える(ステップS10)。これにより、ステップS6の処理によって値が決定された未知画素については、決定された値が既知の値として設定され、この未知画素は、この時点で既知画素となる。一方、ステップS7の処理によって値の決定されなかった未知画素については、書き換えがなされず、未知画素のまま残る。

【0017】次いで、ステップS10の書き換え処理によって、未知の画素がなくなったか否か(あるいは所定個数以下となったか否か)、または、閾値kが最小の値 $k_0$ になったか否かを判断する(ステップS11)。この結果、閾値kが最小の値 $k_0$ になっておらず、未知画素がまだ残っているときには、これら未知画素に対する値を決定するために、閾値kを小さくし(ステップS12)、再びステップS3に戻る。なお、この際、残っている未知画素について、新たに番号jを付す。このようにして、段階的に、未知画素に対する値を決定し、ステップS11において、未知画素がなくなったか、あるいは未知画素の個数が所定個以下となったときに、または、閾値kが最小の値 $k_0$ になったときに、処理を終了する。

【0018】図4(a)乃至(d)は、上記画像処理の具体例を説明するための図である。この具体例では、近傍画素として、図4(a)に示すような $3 \times 3$ の大きさのマスク $M_{11}$ が定義されている。すなわち、このマスク $M_{11}$ の中心を未知画素に設定し、これにより、マスクの8個の部分 $M_{11}$ ,  $M_{12}$ ,  $M_{13}$ ,  $M_{21}$ ,  $M_{22}$ ,  $M_{23}$ ,  $M_{31}$ ,  $M_{32}$ がマスクの中心 $M_{22}$ に設定された未知画素の近傍画素となる。また、この場合、演算の種類が例えば“平均値”に設定され、閾値kの初期値が例えば“7”に設定されているとする。いま、処理対象領域PCUが図4(b)のように特定されている場合、この処理対象領域PCU内の各画素#1~#5は、当初、未知の値のものとなっているので、マスクMの中心 $M_{22}$ を各画素#1~#5に順

次に設定し、処理を行なう。例えば、マスク $M_{11}$ の中心 $M_{22}$ を画素#1に設定すると、 $M_{11}$ ,  $M_{12}$ ,  $M_{13}$ ,  $M_{21}$ ,  $M_{22}$ ,  $M_{23}$ ,  $M_{31}$ ,  $M_{32}$ の7個の画素は、既知領域の画素であり、既知の値をもっており、また、 $M_{23}$ の部分の画素は未知画素であるので、8個の近傍画素のうち、既知の値をもっている既知画素は7個である。従って、閾値k(=7)と同じであり、閾値k(=7)より小さくないので、未知画素#1については、7個の既知画素の値に基づいて値が決定される。例えば、この値は、7個の既知画素の値の平均値として決定される。

【0019】同様に、マスクの中心 $M_{22}$ を画素#5に設定すると、この場合も、8個の近傍画素のうち、7個が既知の値をもっているため、未知画素#5の値が決定される。

【0020】これに対し、マスクの中心 $M_{22}$ を画素#2, #3, または#4に設定すると、 $M_{11}$ ,  $M_{12}$ ,  $M_{13}$ ,  $M_{21}$ ,  $M_{22}$ ,  $M_{23}$ の部分の画素は、既知の値をもっており、 $M_{31}$ ,  $M_{32}$ の部分の画素は未知画素であるので、8個の近傍画素のうち、既知の値をもっている画素は6個である。従って、閾値k(=7)よりも小さいので、未知画素#2, #3, #4については、この段階では値が決定されない。

【0021】このようにして、閾値kを“7”として、全ての未知画素#1~#5についての処理を行なったとき、2つの画素#1, #5について値が決定されたので、画素#1, #5については、決定された値が既知の値として設定される。すなわち、このとき、画素#1, #5は、既知の値をもつ既知画素となり、図4(c)に示すように3つの画素#2, #3, #4が未知のまま残る。

【0022】次の段階では、閾値kを“7”から“6”に変更し、マスク $M_{11}$ の中心 $M_{22}$ を未知の画素#2, #3, #4に順次に設定し、処理を行なう。この場合、マスクの中心 $M_{22}$ を画素#2, または#4に設定すると、8個の近傍画素のうち、7個が既知の値をもっており、閾値k(=6)よりも大きいので、画素#2, #4の値が、7個の既知画素の値に基づいて決定される。同様に、マスクの中心 $M_{22}$ を画素#3に設定すると、8個の近傍画素のうち、6個が既知の値をもっており、閾値k(=6)と同じであり、閾値k(=6)よりも小さくないので、画素#3の値が、6個の既知画素の値に基づいて決定される。

【0023】このようにして、閾値kを“6”として、全ての未知画素#2~#4についての処理を行なったとき、全ての画素#2~#4について、値が決定され、既知の値として設定される。この例では、この段階で、図4(d)に示すように、全ての画素が既知の値をもつ既知画素となり、未知画素が残っていないので、処理を終了する。

【0024】画像処理部2における上述した処理の第1

の特徴(利点)は、処理対象領域PCU内の画素の値のみを決定することにある。すなわち、画像PC中の全ての画素に対する処理を行なうのではなく、画像PC中の一部の領域の画素に対してだけ処理を行なうので、画像処理時間を従来に比べ短縮することができ、また、処理を行なう必要のない既知領域PCK内の画素については処理を行なわないので、全体の画像品質を劣化させずに済む。

【0025】また、上述した処理の第2の特徴(利点)は、未知画素のうち、最も信頼性良く値が決定されうる画素から順に、値を決定することである。すなわち、図4(b)の例では、画素#1、#5は、既知の値をもつ近傍画素数が画素#2~#4に比べて多く、最も信頼性良く値が決定されるものであるため、最初に値が決定される。

【0026】また、上述した処理の第3の特徴(利点)は、未知画素の値が決定され、この未知画素が既知の値をもつ既知画素となるとき、次の段階で、この画素を近傍画素の既知の値をもつ画素の仲間に加えて、未知のまま残っている画素の値を決定することにある。すなわち、図4(c)の例では、例えば、画素#2の値を決定するのに、前の段階で既知の値となった画素#1の値を加味することにある。これにより、各画素の連続性を確保することができる。

【0027】また、上述した処理の第4の特徴(利点)は、画像の種類に依らず、全ての種類の画像を処理対象とすることができることにある。すなわち、近傍画素の定義、閾値の初期値、演算の種類を適宜設定することにより、画像が文書画像のような2値画像であっても、また、写真画像や絵図画像のような多値画像であっても、画像の修復や推定を良好に行なうことができることにある。なお、上述の具体例では、画像の種類について何ら言及していないが、画像が多値画像である場合には、近傍画像のうち既知の値をもつ画素の値の演算値(例えば平均値)そのものを、未知画素の値として決定することができる。また画像が2値画像である場合には、近傍画像のうち既知の値をもつ画素の値の演算値(例えば平均値)が“0.5”以下のときには“0”を未知画素に対する値として決定し、また、演算値(例えば平均値)が“0.5”以上のときには“1”を未知画素に対する値として決定することができる。

【0028】ところで、閾値初期設定部5において、設定される閾値kの初期値としては、例えば、前述のように近傍画素数に応じた最も大きな値に設定することができる。前述の例では、近傍画素数が“8”である場合、閾値kの初期値を“7”に設定することができる。このように、閾値kの初期値を近傍画素数に応じた最も大きな値に設定し、閾値kの最も大きいところから処理を開始することで、処理対象領域PCU内の各画素の値を最も信頼性のあるものから順に決定することができる。

【0029】しかしながら、処理対象となる領域の大きさ、形状によっては、閾値kの初期値を最も大きな値に設定せずとも、画像をある程度信頼性良く良好に処理(例えば修復)することができる場合もある。例えば、処理対象領域PCUの大きさが小さいとき、あるいは処理対象領域PCUの形状が細長いものでないときには、領域PCUの大きさが大きい場合、あるいは領域PCUの形状が細長い場合に比べて、閾値kの初期値を低く設定し、この領域PCUの画素に対する処理が最も信頼性高くなされなくとも、画像全体の品質に対するこの領域PCU内の画素の影響は少ないと考えられる。このような場合にも、閾値kの初期値を常に近傍画素数に応じた最も大きな値に設定し、閾値kの最も大きな値のところから処理を開始すると、処理量が増加し、処理速度が低下するという弊害がかえって生ずる。このような弊害を防止するため、閾値初期設定部5は、処理対象領域PCUの大きさ(すなわち処理対象領域PCU内の画素数)および/または、処理対象領域PCUの形状に応じて、閾値kの初期値を最適なものに設定するのが良い。

【0030】図5は閾値初期設定部5の構成例を示す図である。図5を参照すると、閾値初期設定部5は、閾値kの初期値を決定するための情報を保持するテーブルTBLと、処理対象領域PCUの大きさに関する情報および/または処理対象領域PCUの形状に関する情報を領域情報として検出する領域情報検出部8と、領域情報検出部8で検出された領域情報に対応する閾値kの初期値をテーブルTBLを参照して自動的に決定する初期値決定部9とを有している。

【0031】ここで、領域情報検出部8は、処理対象領域PCUの大きさに関する情報として、画像PC全体の大きさ(すなわち画像PC全体の画素数)に対する領域PCU内の画素数Nの割合f(%)を用いることができる。なお、処理対象領域PCU内の画素数Nは、前述のように領域特定部1によって処理対象領域PCUを特定する際に得られたものを利用することができる。

【0032】また処理対象領域PCUの形状については、領域特定部1によって例えば図6(a)に示すような処理対象領域PCUが特定されたとき、図6(b)のようにこの領域PCUを切り出し、切り出した矩形RCの例えば縦方向iと横方向jの長さの比、すなわち縦横比または横縦比 $RT_{ij}$ ( $=dj/di$ または $di/dj$ )を求め、この縦横比または横縦比 $RT_{ij}$ に基づき検出することができる。すなわち、領域PCUがどの程度の細長さの形状のものであるかを検知することができる。

【0033】また、テーブルTBLは、図7(a)、図7(b)、または図7(c)のように構成することができる。図7(a)の例では、テーブルTBLには、画像PC全体の画素数に対する処理対象領域PCU内の画素数の割合f(%)と、割合fに対応する閾値kの初期値とが設定されている。また、図7(b)の例では、テーブルTBL

Lには、領域PCUの縦横比または縦横比 $RT_{ij}$ と、 $RT_{ij}$ に対応する閾値 $k$ の初期値とが設定されている。なお、図7(b)において、 $RT_{ij}$ は、縦横比( $d_j/d_i$ )または横縦比( $d_i/d_j$ )のいずれか大きい値のものとし、従って、 $RT_{ij}$ は、“1”以上であるとする。また、図7(c)の例では、テーブルTBLには、領域PCU内の画素数の割合 $f$ と、領域PCUの縦横比または横縦比 $RT_{ij}$ と、 $f$ と $RT_{ij}$ とに対応する閾値 $k$ の初期値とが設定されている。

【0034】次に閾値初期設定部5が上記のような構成になっているときの処理の具体例を説明する。いま、テーブルTBLが例えば図7(c)に示すようなものとなっている場合、画像PC中の処理対象領域PCUが例えば図8のように特定されるときには、画像PC全体の画素数に対する処理対象領域PCU内の画素数の割合 $f$ (%)がかなり大きいので、初期値決定部9は、閾値 $k$ の初期値を例えば“7”に決定する。これにより、この大きな領域PCU内の画素の値を極めて信頼性高く決定することができる。

【0035】これに対し、画像PC中の処理対象領域PCUが例えば図9のように特定されるときには、画像PC全体の画素数に対する処理対象領域PCU内の画素数の割合 $f$ (%)が小さく、また、縦横比 $RT_{ij}$ も小さいので、初期値決定部9は、閾値 $k$ の初期値を例えば

“4”に決定する。これにより、演算部7は、この小さな領域PCU内の画素に対する処理を閾値 $k$ の小さな値“4”のところから開始する。なお、閾値 $k$ の小さな値“4”のところから開始する場合には、領域PCU内の画素の値は差程信頼性良くは決定されないが、処理対象領域PCUの大きさが小さいものでありまた、その形状が細長いものではないので、領域PCU内の画素の値が差程信頼性良くは決定されない場合でも、画像PC全体の品質に対する影響は少なく、その反面、処理量を減らし、処理の高速化を図ることができる。

【0036】また、画像PC中の処理対象領域PCUが例えば図10のように特定されるときには、画像PC全体の画素数に対する処理対象領域PCU内の画素数の割合 $f$ (%)は図9と同様に小さいが、縦横比 $RT_{ij}$ が大きくなるので、初期値決定部9は、閾値 $k$ の初期値を例えば“5”に決定する。これにより、図10の領域PCU内に対する処理を、図9の領域PCU内に対する処理と比べ、閾値 $k$ のより大きい値“5”のところから開始する。この結果、細長い形状の領域PCU内の画素の値を図9の場合に比べてより信頼性良く決定できるとともに、図8の場合に比べて、処理量を減らし、処理の高速化を図ることができる。

【0037】このように、本発明では、閾値 $k$ の初期値を処理対象領域の大きさ、形状に応じた最適なものに自動的に決定することにより、処理の最適化を図ることができる。

【0038】なお、上記の例では、1つの画像PC中に1つの処理対象領域PCUが存在する場合について説明したが、図11に示すように、1つの画像PC中に複数の(例えば2つの)処理対象領域PCU1、PCU2が存在する場合にも、各領域PCU1、PCU2について、それぞれ、本発明を同様に適用することができる。この際、画像PC全体の画素数を $N_0$ とし、領域PCU1内の画素数を $N_1$ とし、領域PCU2内の画素数を $N_2$ とすると、閾値 $k$ の初期値を決定するときに用いられる画像PC全体の画素数 $N_0$ に対する割合 $f$ (%)として、全ての処理対象画素数( $N_1 + N_2$ )の画像PC全体の画素数 $N_0$ に対する割合 $(N_1 + N_2)/N_0$ を用いることが考えられる。すなわち、領域PCU1内の各画素に対する処理を行なうときと、領域PCU2内の各画素に対する処理を行なうときとのいずれの場合にも、閾値 $k$ の初期値としては、割合 $f = (N_1 + N_2)/N_0$ により決定された同じものを用いることが考えられる。領域PCU1と領域PCU2とが互いに同程度の大きさのものである場合には、割合 $f = (N_1 + N_2)/N_0$ によって決定された同じ閾値 $k$ の初期値を用いることも可能であるが、図11の例のように、領域PCU1と領域PCU2とが互いに異なる大きさである場合には、これらに一律に同じ閾値 $k$ の初期値を用いることは好ましくない。

【0039】従って、1つの画像PC中に複数の(例えば2つの)処理対象領域PCU1、PCU2が存在する場合、各領域PCU1、PCU2ごとに、割合 $f$ を定めるのが良い。すなわち、領域PCU1に対する閾値 $k$ の初期値 $k_1$ については、この領域PCU1内の画素数 $N_1$ の画像PC全体の画素数 $N_0$ に対する割合 $f_1 = N_1/N_0$ に基づいて決定し、また、領域PCU2に対する閾値 $k$ の初期値 $k_2$ については、この領域PCU2内の画素数 $N_2$ の画像PC全体の画素数 $N_0$ に対する割合 $f_2 = N_2/N_0$ に基づいて決定するのが良い。このように、各領域PCU1、PCU2それぞれに最適な閾値 $k$ の初期値 $k_1$ 、 $k_2$ を決定することで、各領域PCU1、PCU2の大きさ、形状が互いに異なっても(より具体的には、画像PC中において欠陥画素の分布が一樣でない場合にも)、各領域PCU1、PCU2の各画素に対し最適な処理を行なうことができる。

【0040】また、上記各例では、領域PCU(あるいはPCU1、PCU2)の大きさに関する情報として、画像PC全体の画素数に対する領域PCU(あるいはPCU1、PCU2)内の画素数の割合 $f$ (%) (あるいは $f_1$ (%)、 $f_2$ (%))を用いたが、領域PCU(あるいはPCU1、PCU2)の大きさに関する情報として、領域PCU(あるいはPCU1、PCU2)内の画素数そのものを用いても良い。すなわち、画像PC全体が非常に大きいものであって、例えば、領域PCUの大きさ(画素数)が画像PC全体に比べればかなり小さいがそれ自体でかなり大きい場合、領域PCU内の画素の値は、信頼



性高く決定されるのが望ましいが、画像PC全体の画素数に対する領域PCU内の画素数の割合 $f$ により閾値 $k$ の初期値を決定すると、この初期値は領域PCUが画像PC全体と比べ小さな大きさのものであると判断されて低く設定され、従って、領域PCU内の画素の値を信頼性良く決定することができなくなる。このような場合には、領域PCUの大きさに関する情報として、領域PCU内の画素数そのものをを用いるのが良い。

【0041】また、上述の例では、領域PCUの形状に関する情報を、領域PCUの切出矩形RCの縦横比または縦横比 $R_{Tj}$ として検出しており、このような検出の仕方は、領域PCUが図12(a)または図12(b)のように、縦方向 $i$ に沿ってまたは横方向 $j$ に沿って細長い形状である場合には有効である。しかしながら、領域PCU例えば図12(c)に示すように斜めに細長いような場合、上記の検出の仕方では、これが細長い形状であると検出することができない。従って、この場合には、領域PCUの形状に関する情報として、矩形RC内の画素数 $N_i$ に対する領域PCU内の画素数 $N$ の比 $R_{T'}$ を求め、この比 $R_{T'}$ に基づき、領域PCUが細長い形状であるか否かを検出することができる。すなわち、図12(c)の場合に、比 $R_{T'}$ が小さい程、領域PCUが細長い形状であると検出できる。

【0042】また、図12(a)、(b)、(c)のいずれの場合にも、領域PCUが細長い形状であると検出するために、領域PCUの形状に関する情報として、矩形RCの縦横比または横縦比 $R_{Tj}$ とともに、矩形RC内の画素数 $N_i$ に対する領域PCU内の画素数 $N$ の比 $R_{T'}$ を求め、比 $R_{Tj}$ と、比 $R_{T'}$ との両方に基づき、領域PCUが細長い形状であるか否かを検出するのが良い。これによれば、例えば、図12(a)、(b)の場合には、比 $R_{T'}$ は大きい、比 $R_{Tj}$ が大きくなるので、細長い形状と検出でき、また、図12(c)の場合には、比 $R_{Tj}$ は小さい、比 $R_{T'}$ が小さくなるので、領域PCUが細長い形状であると検出することができる。

【0043】上述したような本発明の画像処理装置は、処理対象となる領域が特定できるものであれば、種々の用途に適用可能である。なお、適用される用途が異なるごとに、処理対象領域の特定の仕方も異なるので、画像処理装置の領域特定部1は、適用される用途に適した構成をとる必要がある。

【0044】本発明の画像処理装置の適用例として、ファクシミリ、複写機あるいはイメージスキャナ等の画像読取システムで読取った画像を修復する場合が考えられる。図13は本発明の画像処理装置が画像修復装置として組み込まれた画像読取システムの構成例を示す図である。図13を参照すると、この画像読取システムは、原稿10が載置される原稿台11と、コンタクトガラス12と、原稿10に光を照射する光源13と、原稿10からの反射光を検知し、原稿10の読取画像を得るCCD

素子やラインセンサなどの画像読取デバイス14と、画像読取デバイス14によって読取られた画像(例えば多値画像、あるいは2値画像)を記憶する画像メモリ15と、画像メモリ15に記憶された画像に対し修復処理を施す画像修復装置16とを備えている。

【0045】このような画像読取システムにおいて、処理対象、すなわち修復対象となるべき画像の欠陥は、例えば、コンタクトガラス12(あるいは、図示しないがフィルタ、レンズなどの光学部品)の一部に汚れや傷などがある場合、あるいは、CCD素子やラインセンサなどの画像読取デバイス14に故障がある場合、あるいは原稿10の一部に汚れや傷、しわなどがある場合に生ずる。

【0046】ここで、コンタクトガラス12(あるいは、フィルタ、レンズなどの光学部品)の一部に汚れや傷などがある場合、これらの汚れや傷に対応した欠陥は、これらが取り除かれない限り、画像メモリ15中の一定領域(同じ領域)に常に現われる。コンタクトガラス12(あるいは、フィルタ、レンズなどの光学部品)の一部に存在する汚れや傷などに対応した画像の欠陥を画像修復装置16により修復する場合、まず、画像メモリ15中で画像の欠陥領域、すなわち修復対象領域(処理対象領域)を領域特定部1によって特定する必要があるが、この修復対象領域は、画像読取りを繰返し行なっても、画像メモリ15中の常に同じ位置となるので、例えば、原稿台11に原稿10が載置されていない状態(例えば白紙が載置されている状態)における画像を読取って画像メモリ15にプリスキャン画像として記憶し、画像メモリ15に記憶されたプリスキャン画像から欠陥領域すなわち修復対象領域を特定するよう、領域特定部1を構成することができる。

【0047】例えば、領域特定部1は、画像メモリ15中にプリスキャン画像が多値で記憶されるとする場合、所定濃度値以上の画素からなる領域を修復対象領域として抽出し特定するよう構成することができる。また、画像メモリ16中にプリスキャン画像が2値で記憶されるとする場合、“1”の値をもつ画素からなる領域を修復対象領域として抽出し特定するよう構成することができる。図14は領域特定部1の具体例を示す図であり、図14の例では、領域特定部1は、領域を抽出する領域抽出部17と、抽出した領域内の各画素の位置情報が記憶される領域メモリ18とを有している。

【0048】いま、プリスキャンを行なうことによって、例えば図15(a)に示すようなプリスキャン画像が得られる場合、領域特定部1の領域抽出部17は、図15(a)のプリスキャン画像から2つの修復対象領域PCU1、PCU2を抽出し特定し、各領域PCU1、PCU2内の画素の位置情報を領域メモリ18に格納する。このとき、画像処理部2の閾値初期設定部5は、領域メモリ18に格納された位置情報を参照し、前述したよう

な仕方では、修復対象領域 PCU1 に対する閾値  $k$  の最適な初期値  $k_1$  を設定し、また、修復対象領域 PCU2 に対する閾値  $k$  の最適な初期値  $k_2$  を決定する。このようにしてブリスキャンにより、修復対象領域が抽出し特定され、また、閾値  $k$  の最適な初期値  $k_1$ 、 $k_2$  が決定された後、実際の原稿を読み取り、その画像を画像メモリ 15 に記憶する。図 15(b) は実際の原稿画像の一例であり、図 15(b) と図 15(a) とを比べればわかるように、図 15(b) においても、図 15(a) とほぼ同じ位置に、コンタクトガラス 12 (あるいは、フィルタ、レンズなどの光学部品) の一部に存在する汚れや傷などに対応した欠陥が生じている。画像修復装置の画像処理部 2 は、領域メモリ 18 に格納に格納された領域 PCU1、PCU2 の位置情報を参照し、図 15(b) の原稿画像において、修復対象領域 PCU1、PCU2 内の各画素が値をもっている、この値を何ら考慮せずに、(すなわちこれら各画素の値が完全に未知のものであると仮定した上で)、画像メモリ 15 に記憶されている原稿画像に対し、前述したような処理を修復処理として行なうことができる。例えば、最初、原稿画像において修復対象領域 PCU1 の各画素に対して、閾値  $k$  の最適な初期値  $k_1$  のところから修復処理を行い、次いで、修復対象領域 PCU2 内の各画素に対して、閾値  $k$  の最適な初期値  $k_2$  のところから修復処理を行なう。

【0049】このようにして、コンタクトガラス 12 (あるいは、フィルタ、レンズなどの光学部品) の一部に存在する汚れや傷などによって、画像の一部の領域に欠陥が生じた場合にも、この領域に応じた最適な仕方では、この領域を高品質の状態に迅速に修復することができる。

【0050】また、画像読取デバイス 14 の一部に故障がある場合にも、コンタクトガラス 12 に汚れや傷などがある場合と同様に、画像読取デバイス 14 の一部の故障に対応した欠陥は、図 16(a) あるいは図 16(b) に示すように、画像読取デバイス 14 が交換されない限り、画像メモリ 15 中の一定領域 (同じ領域) に常に現われる。例えば、図 16(a) に示すように白抜けとして現われたり、図 16(b) に示すように黒線として現われたりする。なお、図 16(a)、(b) は、画像読取デバイス 14 が CCD である場合の画像である。

【0051】従って、画像修復装置 16 の領域特定部 1 を、コンタクトガラス 12 上の汚れや傷に対応した欠陥領域を特定する図 14 に示したと同様の構成のものにすることで、画像読取デバイス 14 の一部に故障がある場合の画像中の欠陥領域を特定することができ、特定した欠陥領域の各画素を画像処理部 2 により修復することができる。換言すれば、画像修復装置 16 を図 14 の構成とすることで、コンタクトガラス 12、画像読取デバイス 14 などを含む画像読取システム全体の欠陥に対応した画像の欠陥領域を同時に特定し、これらを 1 つの画像

修復装置 16 で修復することができる。

【0052】また、画像読取デバイス 14 の一部に故障があるときに生ずる画像の欠陥領域だけを特定する場合には、領域特定部 1 を図 17 に示したような構成のものにすることができる。すなわち、図 17 の構成では、領域特定部 1 は、CCD やラインセンサなどの画像読取デバイス 14 の個々の素子から出力される信号レベルを検出し、各素子が正常であるか異常であるかを判定する異常検出部 21 と、異常と判定された素子に対応する画像メモリ 15 上での画素の位置情報を記憶する領域メモリ 22 とを有している。

【0053】一般に、画像読取デバイス 14 の一部の素子が電気的に壊れると、この素子から出力される信号レベルは、常時異常なレベルとなり、画像中に図 16(a)、図 16(b) に示したような欠陥が生ずる。従って、領域特定部 1 が図 17 の構成となっている場合には、異常検出部 21 により信号レベルが異常であると検出された素子に対応する画素からなる領域を修復対象領域として特定し、その位置情報を領域メモリ 22 に格納する。そして、画像処理部 2 は、領域メモリ 22 に格納された位置情報を参照し、まず、閾値  $k$  の初期値を最適なものに決定し、しかる後、最適に決定された閾値  $k$  の初期値のところから画像修復を行い、図 16(a)、図 16(b) のような欠陥のある画像を、この欠陥に応じた最適な仕方では、良好に修復することができる。

【0054】なお、画像修復装置 16 のもつ上記機能を画像読取デバイス 14 自体にもたせることもできる。この場合、画像読取デバイス 14 の一部が破損しても、画像読取デバイス 14 自体でこの破損した部分に対応する画像を良好に修復でき、画像読取デバイス 14 からは、画像読取デバイス 14 の一部が破損していないときと同様の良好な品質の画像が出力されるので、素子の寿命を見かけ上延ばすことができる。

【0055】一方、原稿 10 の一部に汚れや傷、しわなどがある場合、原稿 10 は、コンタクトガラス 12 や画像読取デバイス 13 などのように、画像読取システムに固定されたものでないで、原稿 10 を読取ったときに、原稿 10 上の汚れや傷、しわなどは、画像メモリ 15 中において常に同じ領域に現われる性質のものではない。

【0056】従って、原稿 10 上の汚れや傷、しわなどによる画像の欠陥を画像修復装置 16 により修復する場合、画像修復装置 16 は、例えば図 18 のように構成される。すなわち、図 18 の画像修復装置 16 では、領域特定部 1 は、画像メモリ 15 に記憶された画像を表示する表示部 25 と、表示部 25 に表示された画像のうち所定の領域をオペレータに指示させるためのマウスなどのポインティングデバイス 26 と、ポインティングデバイス 26 によって指示された領域内の各画素の位置情報が記憶される領域メモリ 27 とを有している。

10

20

30

40

50

【0057】このような構成では、原稿10を読取ったとき、画像メモリ15に記憶された原稿画像が例えば図19(a)のように表示部25に表示される。この表示を見て、オペレータは、原稿10上の汚れや傷、しわなどによる部分を特定し、その部分を図19(b)に符号PCUで示すように、マウスなどのポインティングデバイス26によって指示する(例えば囲む)。これにより、ポインティングデバイス26によって指示された(囲まれた)領域PCU内の各画素の位置情報が領域メモリ27に格納される。そして、画像処理部2は、領域メモリ27に格納された位置情報を参照し、先ず、閾値kの初期値を最適なものに決定し、しかる後、最適に決定された閾値の初期値のところから領域PCU内の各画素に対し、前述のような仕方では画像修復を行なうことができる。なお、図18の例では、処理対象となる領域をオペレータに指示させるようになっているが、汚れや傷が一定の形状、大きさのものである場合、処理対象となる領域も一定の形状、大きさのものとなるので、この場合には、パターン認識技術を用いてオペレータの指示によらずに領域を特定することも可能である。

【0058】このように、本発明の画像処理装置は、画像読取システムにおいて読取画像中に生ずるあらゆる種類の欠陥に対し、この欠陥の大きさ、形状に応じた最適な仕方では、これを良好に画像修復することができる。

【0059】上記適用例は、原稿の読取画像(2次元画像)に対して画像の修復を行なう場合であるが、本発明は、本願と同一の出願人により本願と同日付で出願された特許出願(整理番号9303862)に記載されていると同様に、対象画像として、2次元画像のみならず、CT画像や核磁気共鳴画像などの3次元画像あるいは動画

【0060】また、ファクシミリの画像伝送時に、例えば、回線等にノイズなどが混入するときに生ずるファクシミリ受信画像の欠陥の修復などにも適用可能である。

【0061】また、上記各例では、演算部7において、処理対象領域PCU内の未知の画素を完全に未知の値のものとし(未知画素が不確かな値をもっている場合、この値を何ら考慮せずに)、近傍画素のうちの既知画素の値だけに基いて、未知画素の値を決定するようにしているが、未知画素が完全には未知でなく不確かな値をもっている場合、この情報をも考慮した演算処理がなされることによって、より高品質の画像に修復することができることもある。

【0062】従って、演算部7として、未知画素が不確かな値をもっている場合、前述の処理例のようにこの未知画素の値が完全に未知であると仮定し近傍画素のうちの既知画素の値だけによって決定し、しかる後、このように決定した値に、この未知画素が本来もっている不確かな値を加味して、この未知画素の値を最終的に決定す

るよう構成することもできる。

【0063】すなわち、未知画素が本来もっている不確かな値を $p(x)$ とし(ここで、 $x$ は未知画素の位置を表わし、 $p$ は例えば濃度値を表わしている)、また、この未知画素が完全に未知であると仮定し近傍画素のうちの既知画素の値だけによって決定した値を $q(x)$ とすると、 $p(x)$ と $q(x)$ との算術演算の結果 $r(x)$ を、この未知画素の値として最終的に決定するようにすることもできる。

【0064】 $p(x)$ と $q(x)$ との算術演算の具体例として、例えば次式のように、 $p(x)$ と $q(x)$ の平均値 $\langle p(x) + q(x) \rangle$ をとることもできる。

【0065】

【数1】  $r(x) = \langle p(x) + q(x) \rangle$

【0066】あるいは、次式のように、処理対象領域PCUの各画素について得られた値 $p(x)$ のうちで最小のものを $p'(x)$ とすると、 $p(x)$ と $q(x)$ の平均値 $\langle p(x) + q(x) \rangle$ から $p'(x)$ を減算した値 $\{ \langle p(x) + q(x) \rangle - p'(x) \}$ を $r(x)$ とすることもできる。

【0067】

【数2】

$$r(x) = \langle p(x) + q(x) \rangle - p'(x)$$

$$p'(x) = \min_{x \in PCU} \{ p(x) \}$$

【0068】処理対象領域PCU内の未知画素が不確かな値をもっている場合として、例えば、前述の第1の適用例に示したような画像読取システムで読取った画像における欠陥が挙げられる。すなわち、コンタクトガラス12(あるいはフィルタ、レンズなどの光学部品)の一部に汚れや傷がある場合、あるいは、画像読取デバイス14の一部の素子の感度が低い場合、あるいは、原稿の一部に汚れや傷、しわなどがある場合のように、欠陥部分の各画素の値が全く未知ではなく、その信頼性がいくらか低い場合が挙げられる。このような場合に、未知画素の値を完全に未知のものと仮定して、この未知画素の値 $r(x)$ を $p(x)$ として決定してしまうと、この未知画素が本来もっている情報が何ら生かされない。これに対し、上述のように、未知画素の値 $r(x)$ を $p(x)$ と $q(x)$ との算術演算値とすることによって、より高品質に画像を修復することができる。

【0069】さらに、上記変形例として、未知画素の本来の値 $q(x)$ が信頼性の高いものか否かを判断し、信頼性の高いときには、未知画素の値 $r(x)$ を数1または数2のように決定する一方、信頼性の低いときには、未知画素の本来の値が完全に未知であると、未知画素の値 $r(x)$ を $p(x)$ として決定するようにしても良い。

【0070】また、上述の各例では、処理対象領域PCU内の画素に対して、1画素単位に処理がなされるとしたが、処理を高速に行なうため、例えば図20に示すよ

うに、複数の画素の組を1つの画素とみなし、この画素単位に処理を行なうようにすることもできる。なお、図20の例では、複数の画素の組が $2 \times 2$ の画素からなっているとおり、この場合、近傍画素も複数の画素の組からなるものとして定義することができる。例えば、近傍画素の1つの画素を領域PCU内の画素と同様に、 $2 \times 2$ の画素からなるものとして定義することができる。

【0071】このように、複数の画素の組を1つの画素とみなし、この画素単位に処理がなされる場合には、1画素単位に処理がなされる場合に比べて、処理対象となる画素数を低減することができ、処理を高速に行なうことができる。例えば、 $2 \times 2$ の画素を1つの画素とみなし、この画素単位に処理がなされる場合には、処理速度を4倍に向上させることができる。但し、このような処理を行なうことによって、画像の解像度が劣化する。上記例では、解像度は $1/2$ に劣化する。従って、解像度が劣化しても良いような用途に適用することができる。

【0072】その適用例として、例えば、送信側のファクシミリ装置が400dpiの解像度で画像信号を送る能力を有しているのに対し、受信側のファクシミリ装置が200dpiの解像度の再現能力しか有していない場合である。この場合、受信側のファクシミリ装置では、送信側のファクシミリ装置からの400dpiの画像に対して、前述したような処理を行なう場合、送信側における400dpiの解像度を維持する必要はなく、200dpiの解像度があれば良い。従って、 $2 \times 2$ の画素を1つの画素とみなし、この画素単位に画像修復処理を行なうことにより、処理速度を4倍に向上させ、かつ、受信側に必要な解像度を満たすことができる。

【0073】

【発明の効果】以上に説明したように、請求項1乃至請求項7記載の発明によれば、処理対象領域内の各画素が当初、完全に未知の値あるいは不確かな値をもつ未知画素であるとし、処理対象領域内の特定の未知画素に着目するとき、該未知画素の近傍画素のうちで既知の値をもつ既知画素の個数が所定の閾値よりも大きい場合に限り、近傍画素のうちの既知画素の値を用いて未知画素の値を決定する一方、既知画素の個数が所定の閾値よりも小さいときには、未知画素の値を決定せず未知画素のまま残すようになっており、処理対象領域内の各画素について所定の閾値での前記処理がなされた後、値の決定された未知画素を既知画素とし、また、閾値を変更し、残されている未知画素に対して処理を繰り返して行なう画像処理において、上記閾値の最適な閾値を、処理対象領域の大きさに関する情報、および/または、処理対象領域の形状に関する情報に基づいて決定するようになっているので、処理対象領域の大きさ、形状に応じた最適な処理を行なうことができ、画像を高速にかつ高品質に修復、推定等することができる。

【図面の簡単な説明】

【図1】本発明に係る画像処理装置の構成例を示す図である。

【図2】画像中の処理対象領域と既知領域とを説明するための図である。

【図3】図1の画像処理装置の動作を説明するためのフローチャートである。

【図4】(a)乃至(d)は図1の画像処理装置の処理の具体例を示す図である。

【図5】閾値初期設定部の構成例を示す図である。

【図6】(a), (b)は処理対象領域の形状を検出する仕方を説明するための図である。

【図7】(a)乃至(c)はテーブルの構成例を示す図である。

【図8】閾値初期設定部の処理の具体例を示す図である。

【図9】閾値初期設定部の処理の具体例を示す図である。

【図10】閾値初期設定部の処理の具体例を示す図である。

【図11】1つの画像中に複数の処理対象領域が存在する場合を示す図である。

【図12】(a)乃至(c)は領域PCUの形状を説明するための図である。

【図13】本発明の画像処理装置が画像修復装置として組み込まれた画像読取システムの構成例を示す図である。

【図14】画像処理装置の具体的な構成例を示す図である。

【図15】(a)はブリスキャン画像の一例を示す図、(b)は実際の原稿画像の一例を示す図である。

【図16】(a), (b)は画像読取デバイスの一部の素子に欠陥がある場合の画像の一例を示す図である。

【図17】画像処理装置の具体的な構成例を示す図である。

【図18】画像処理装置の具体的な構成例を示す図である。

【図19】(a), (b)は表示部上での処理対象領域の特定の仕方を説明するための図である。

【図20】(a), (b)は処理対象領域内の画素に対して、複数の画素の組を1つの画素とみなして処理を行なう場合の一例を示す図である。

【符号の説明】

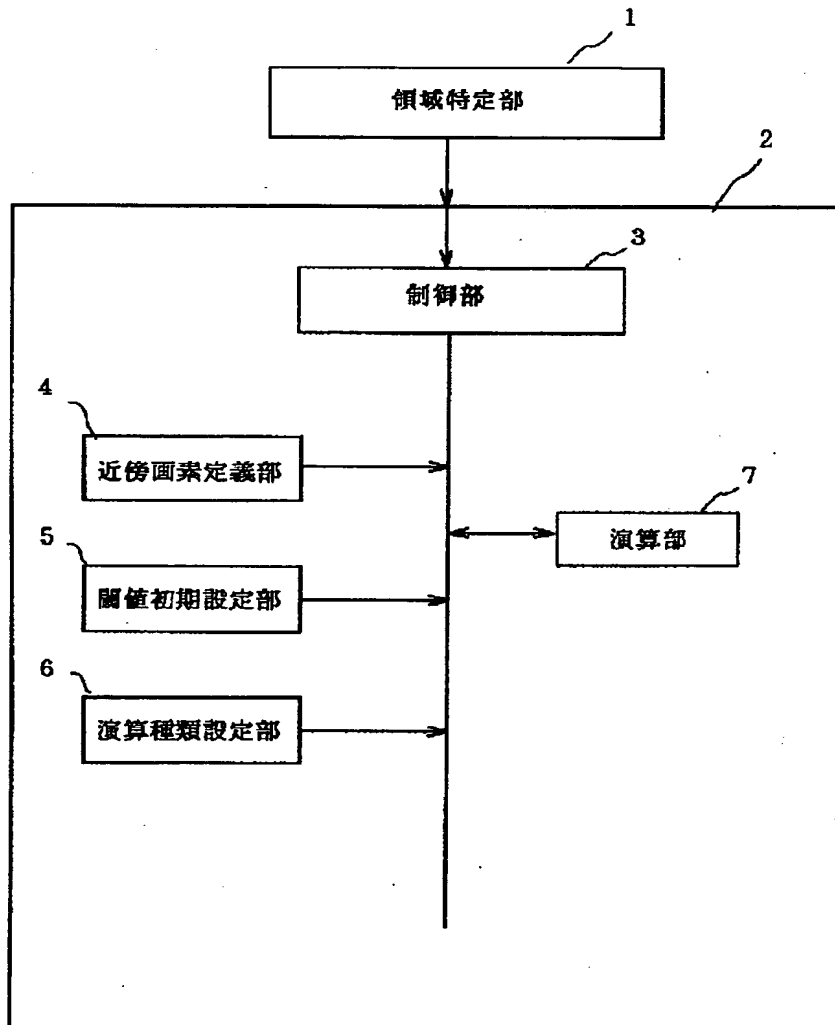
1	領域特定部
2	画像処理部
3	制御部
4	近傍画素定義部
5	閾値初期設定部
6	演算種類設定部
8	領域情報検出部

19  
 9 初期値決定部  
 10 原稿  
 11 原稿台  
 12 コンタクトガラス  
 13 光源  
 14 画像読取デバイス  
 15 画像メモリ  
 16 画像処理装置  
 17 領域抽出部

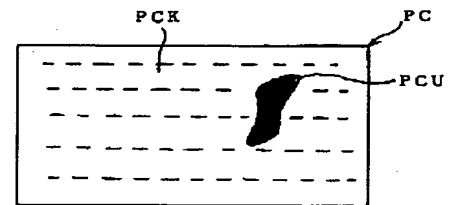
18, 22, 27  
 21  
 25  
 26  
 PC  
 PCU  
 PCK  
 TBL

20  
 領域メモリ  
 異常検出部  
 表示部  
 ポインティングデバイス  
 画像  
 処理対象領域  
 既知領域  
 テーブル

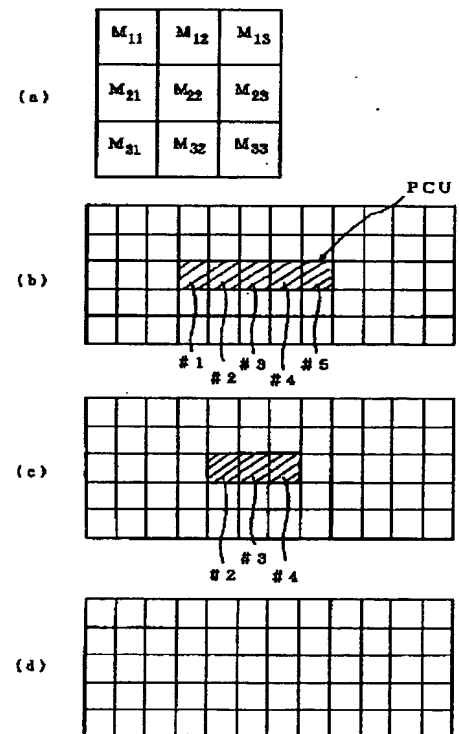
【図1】



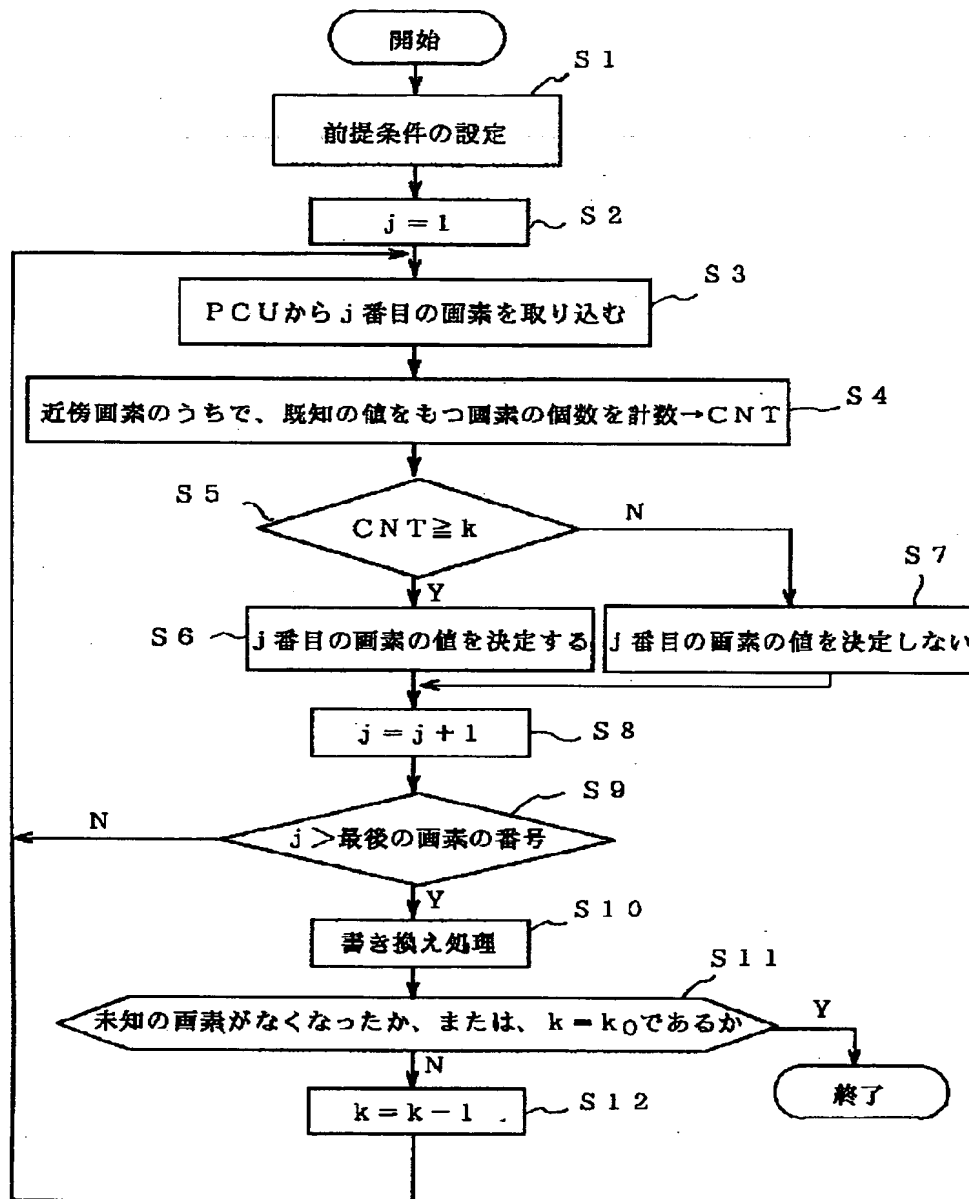
【図2】



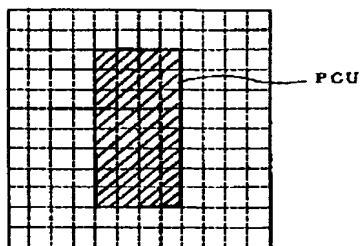
【図4】



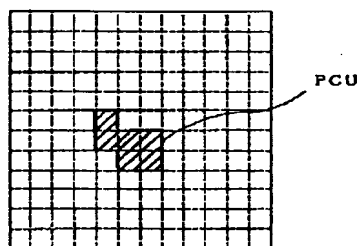
【図 3】



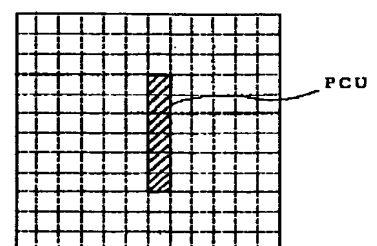
【図 8】



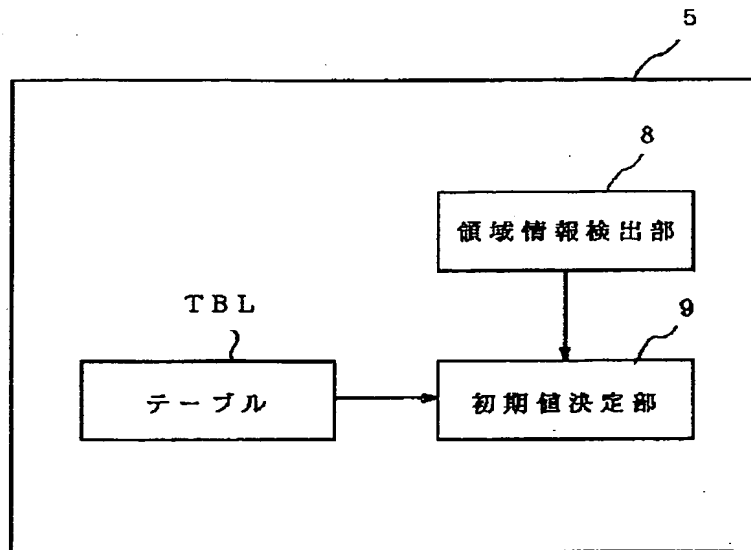
【図 9】



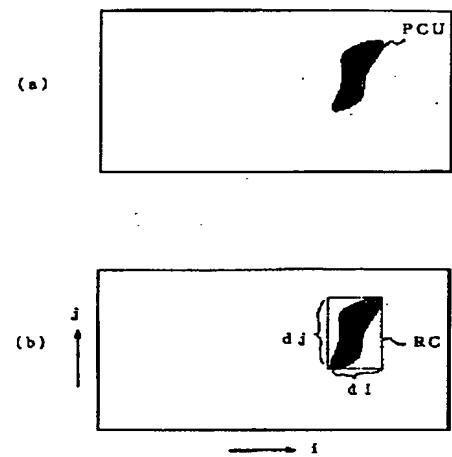
【図 10】



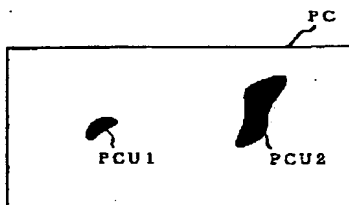
【図 5】



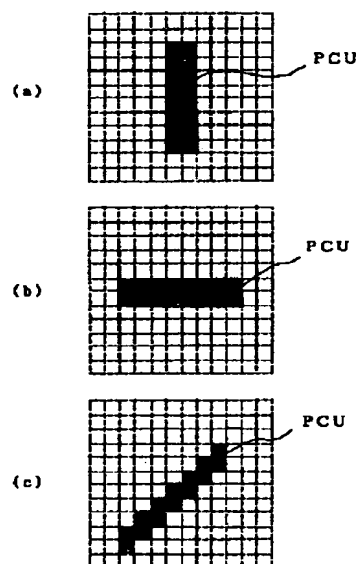
【図 6】



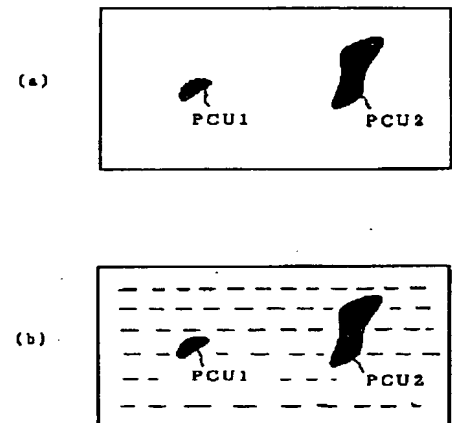
【図 11】



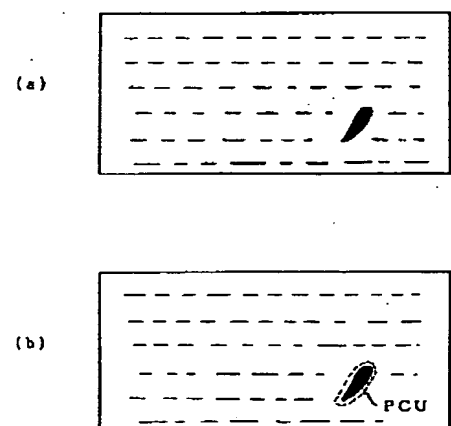
【図 12】



【図 15】



【図 19】



【図7】

(a)

$f(\%)$	閾値 $k$ の初期値
$0 < f \leq 1$	4
$1 < f \leq 2$	5
$2 < f \leq 5$	6
$5 < f \leq 10$	7
$10 < f$	8

(b)

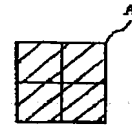
$RT_{ij}$	閾値 $k$ の初期値
$1 \leq RT_{ij} < 1.5$	4
$1.5 \leq RT_{ij} < 2$	5
$2 \leq RT_{ij} < 4$	6
$4 \leq RT_{ij} < 8$	7
$8 \leq RT_{ij}$	8

(c)

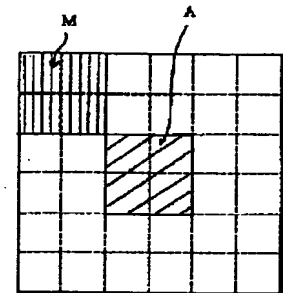
$f(\%)$	$RT_{ij}$	閾値 $k$ の初期値
$0 < f \leq 2$	$1 \leq RT_{ij} < 2$	4
$0 < f \leq 2$	$2 \leq RT_{ij} < 8$	5
⋮	⋮	⋮
$5 < f$	$8 \leq RT_{ij}$	8

【図20】

(a)

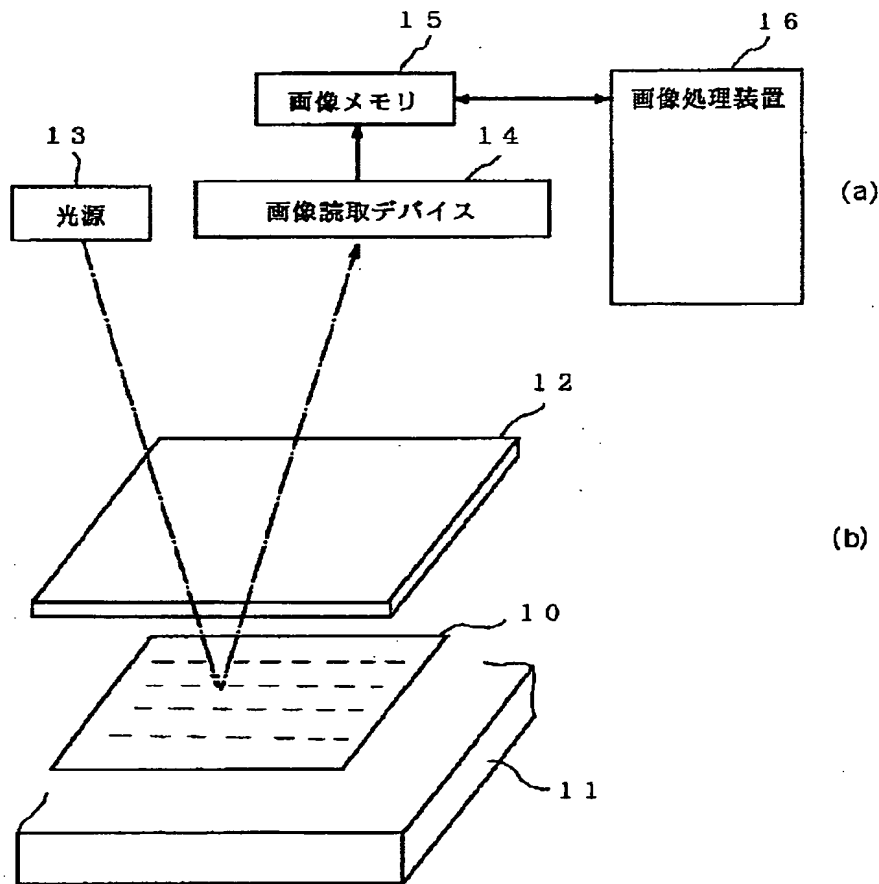


(b)

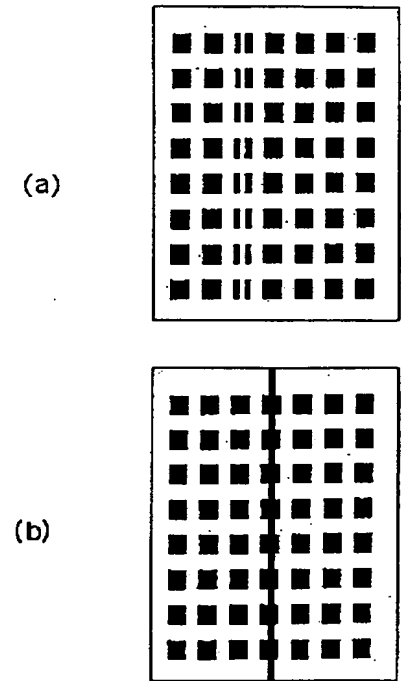




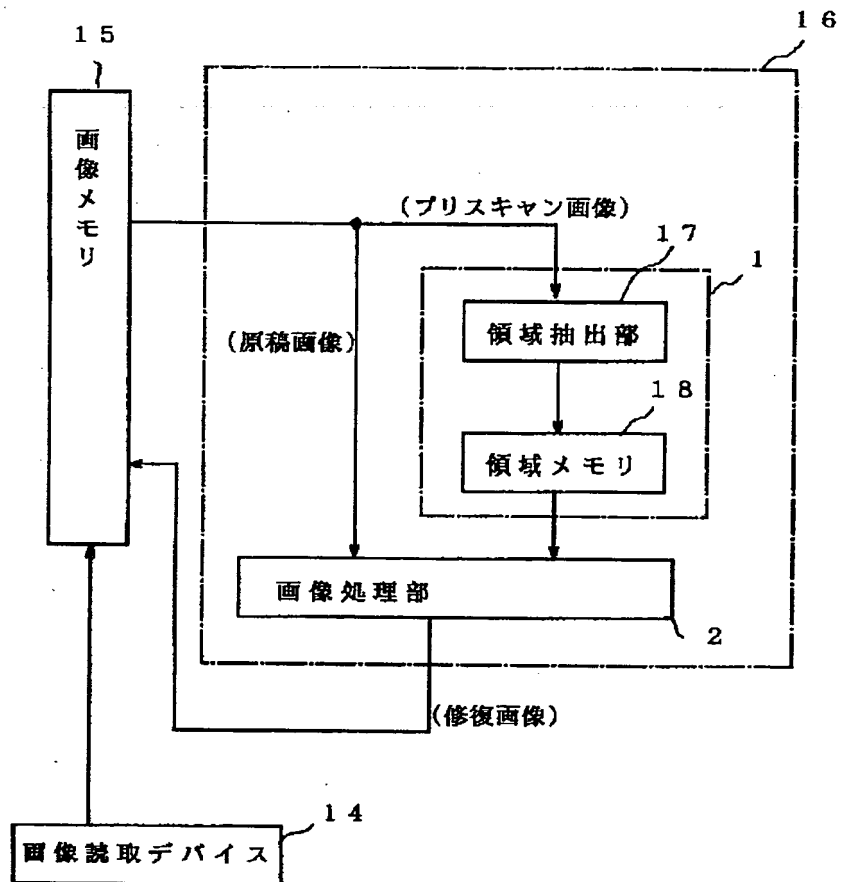
【図13】



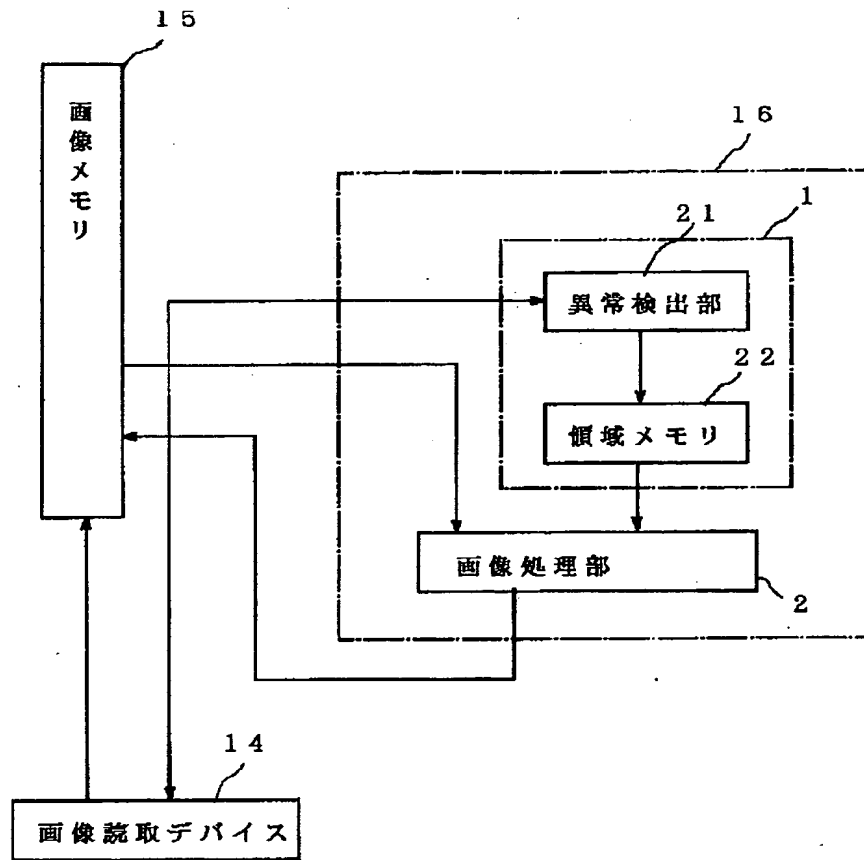
【図16】



【図 14】



【図 17】



```

graph TD
    14[画像読取デバイス] --> 15[画像メモリ]
    15 --> 25[表示部]
    15 --> 2[画像処理部]
    14 --> 2
    2 -- 修復画像 --> 15
    subgraph 16 [ ]
        25 --> 26[ポインティングデバイス]
        26 --> 27[領域メモリ]
        27 --> 2
    end

```

Figure 1 is a block diagram of the image processing system. The diagram shows the flow of data between various components. At the bottom left is the 'Image Reading Device' (14), which sends data to the 'Image Memory' (15). The 'Image Memory' (15) is connected to the 'Display Unit' (25) and the 'Image Processing Unit' (2). The 'Image Processing Unit' (2) receives input from the 'Image Memory' (15) and the 'Image Reading Device' (14), and outputs 'Restored Image' (修復画像) back to the 'Image Memory' (15). Inside a dashed box (16), the 'Display Unit' (25) is connected to the 'Pointing Device' (26), which is connected to the 'Area Memory' (27). The 'Area Memory' (27) is connected to the 'Image Processing Unit' (2). The 'Image Processing Unit' (2) also receives input from the 'Image Memory' (15) and outputs 'Restored Image' (修復画像) back to the 'Image Memory' (15).